

Easy Hacks to get you into:



LibreOffice

The Document Foundation

How beginners (as well as experts) can help ...

Michael Meeks

michael.meeks@suse.com

*“Stand at the crossroads and look; ask for the ancient paths, ask where the good way is, and walk in it, and you will find rest for your souls...” -
Jeremiah 6:16*



The message in a nutshell ...

- **Easy**
(to create & to hack on)
- **Significant**
They make a real impact in the project
- **You can do them**
There is nothing to stop you hacking today
- Lets look at some from easy → hard.

Easy: Translating German Comments ...

- Admittedly much easier if you are a German

```
git clone git://anongit.freedesktop.org/libreoffice/core
```

- Beautiful tool tool help out with this:

```
cd clone
```

```
bin/find-german-comments cppuhelper # random directory
```

```
cppuhelper/source/propshlp.cxx:706:
```

```
Ueber alle Listener iterieren und Events senden
```

```
cppuhelper/source/propshlp.cxx:763:
```

```
Ueber alle Listener iterieren und Events senden
```

```
cppuhelper/source/unour1.cxx:264:
```

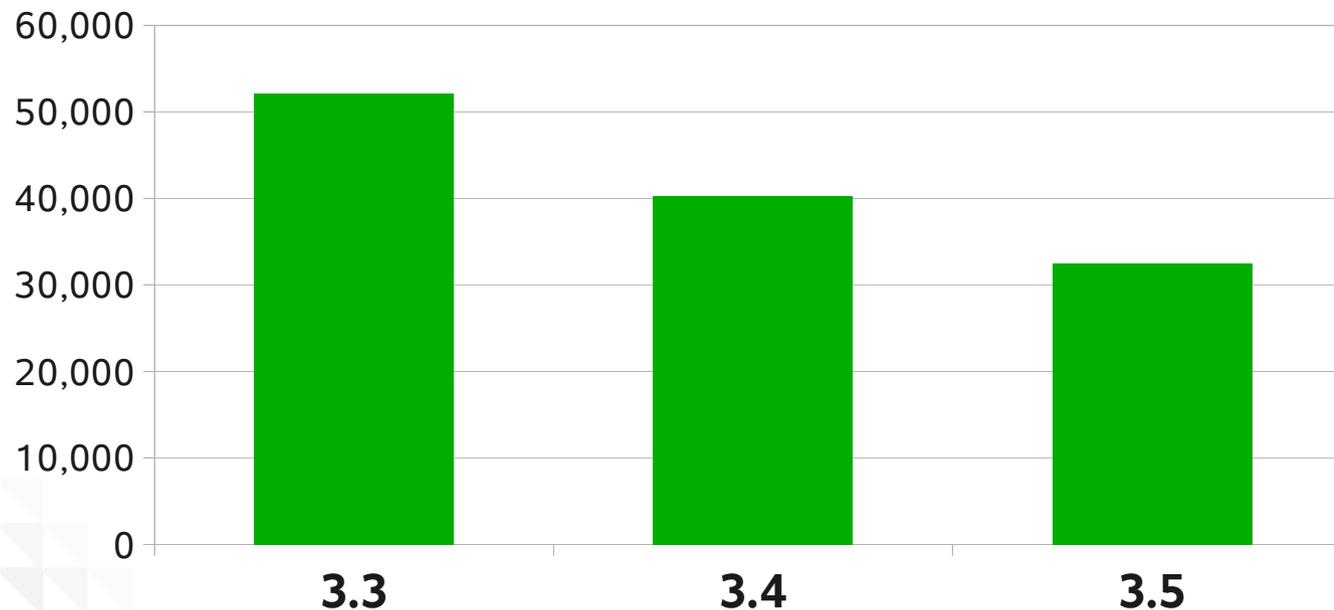
```
c != 0x2F && c != 0x3A && c != 0x3D
```

Only two (identical) strings detected – to clean an entire module

Significant: Translating Comments ...

- ▶ It makes things **significantly** easier for non-Germans ...
- ▶ We removed or translated ~20k lines of these so far
- ▶ We still have 32k to go: done mostly by new guys ...
 - ▶ Thanks to all who helped !

Detected lines of German comment



Thanks to (recent translators):
Philipp Weissenbacher
Luc Castermans
Mike Whiteley
Christina Rossmannith
Winfried Donkers
Michael Stahl
Tim Hardeck
Noel Grandin
Matteo Casalin
Cor Nouws
and many more !

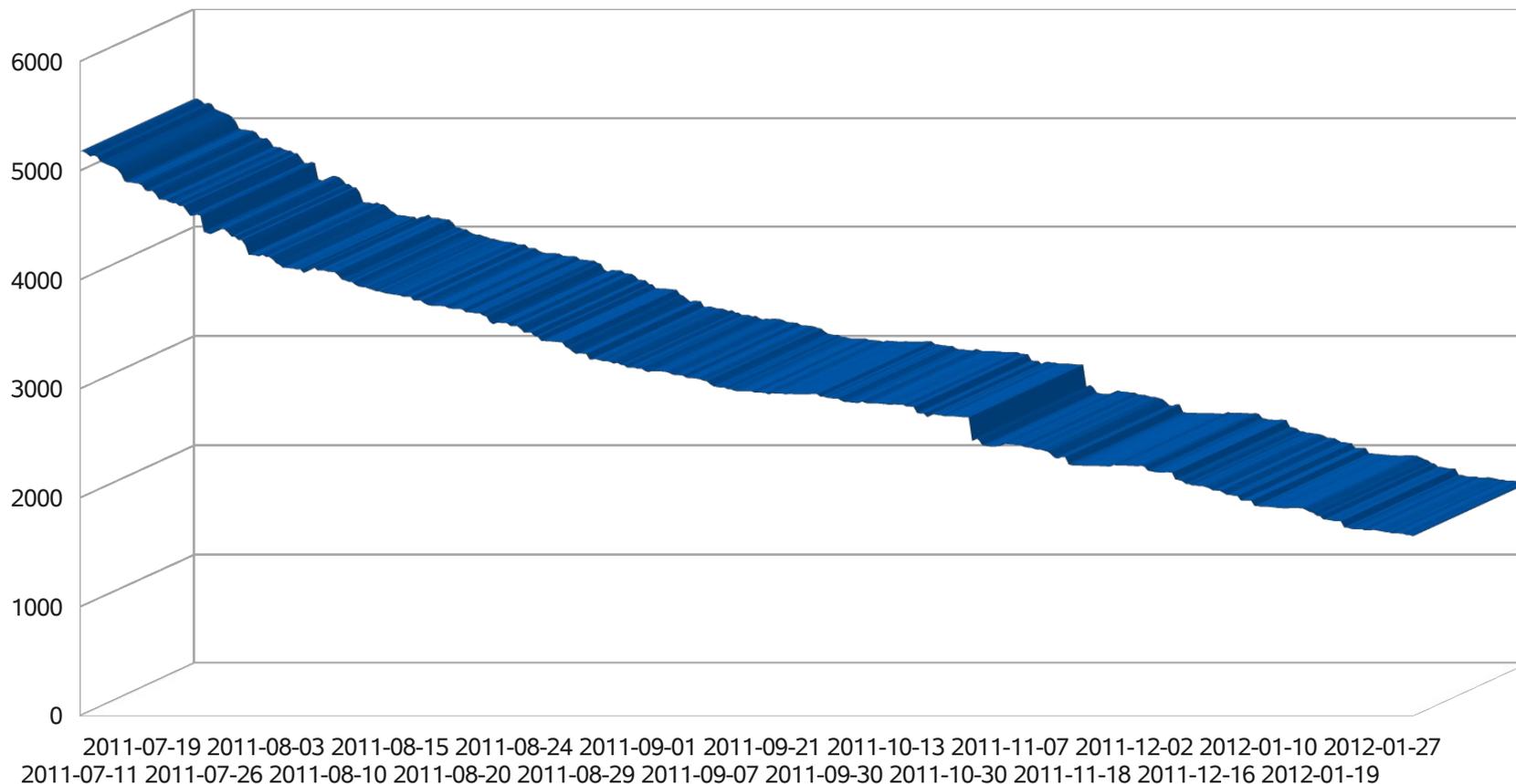
Easy: Removing totally unused code ...

- ▼ Inherited code, has a **big** technical debt: needs paying ...
 - ▼ `git clone git://anongit.freedesktop.org/libreoffice/core`
 - ▼ `cd core`
- ▼ Load up `unusedcode.easy` ...
 - ▼ `EditView::GetSelectionMode() const`
 - ▼ `EditView::SetCursor(Cursor const&)`
 - ▼ `EditView::SetPointer(Pointer const&)`
 - ▼ ...
- ▼ Find these methods:
 - ▼ `git grep EditView::GetSelectionMode`
 - ▼ `editeng/source/editeng/editview.cxx:EESelectionMode`
`EditView::GetSelectionMode() const`
- ▼ Remove the method (don't forget the header)
 - ▼ If it looks -really- useful: `#ifdef TOMORROW / #endif` ...

Significant: Removing totally unused code ...

- ▶ Shrinking the LibreOffice's memory footprint
- ▶ We're making real progress ... 5k- to less than 2k

Remaining unused method count



Easy: Using git ...

- ▼ Git can **sound** scary – but it's not -so- bad ...
- ▼ One off setup your of name / E-mail (we like real ones)
 - ▼ `git config user.name "Jim Bob"`
 - ▼ `git config user.email "jim.bob@easy.com"`
- ▼ Checkout the code:
 - ▼ `git clone git://anongit.freedesktop.org/libreoffice/core`
- ▼ Edit a file you like
 - ▼ `git diff #` will show you your diff
- ▼ `git commit -a #` commit all changes
- ▼ `git format-patch -1 HEAD #` create a patch to E-mail us ...
- ▼ `git pull -r #` get the latest changes from others
- ▼ `git stash #` squirrel away your changes, giving a clean tree.
- ▼ `git stash pop #` get them back again ...

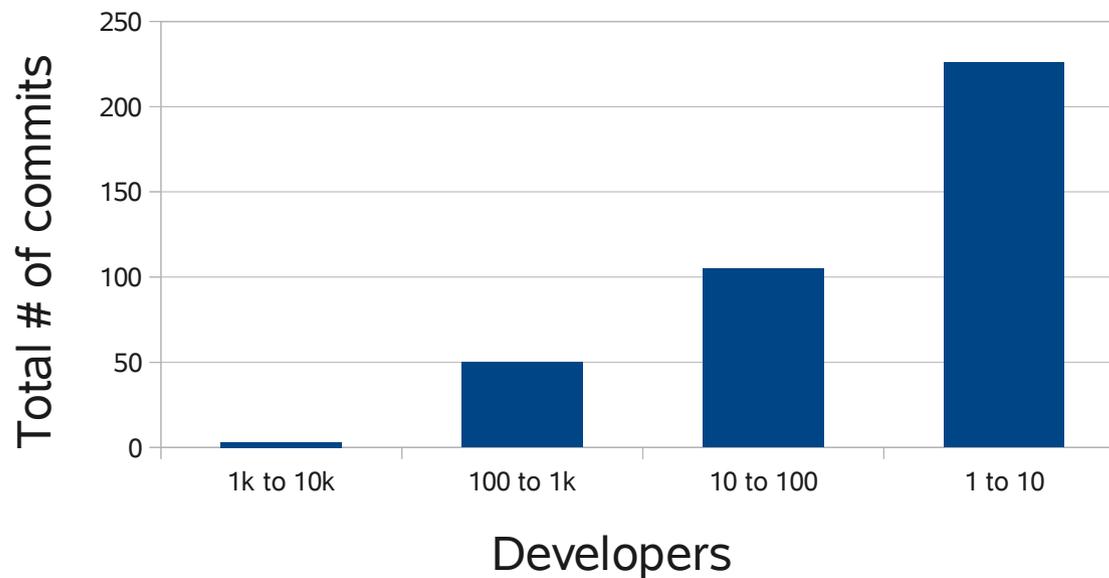
Easy: Removing old, crufty conditionals ...

- ▶ We used to compile with some *horribly* broken compilers
 - ▶ We still have conditionals around the place for them
 - ▶ We should remove all checks for `__GNUC__ < 3`
- ▶ So checkout:
 - ▶ `git clone git://anongit.freedesktop.org/libreoffice/core`
- ▶ Then:
 - ▶ `git grep __GNUC__`
 - ...
 - `cppu/inc/cppu/macros.hxx:`
 - `#if defined(__GNUC__) && (__GNUC__ >= 3)`
 - `sal/osl/unx/file_stat.cxx:`
 - `#if defined(__GNUC__) && (__GNUC__ < 3)`
 - ...
 - ▶ Remove conditionals and associated code

Significant: cleaner code, easier to grep/read

- ▶ Makes it easier to hack on LibreOffice
- ▶ Continue reducing our (bogus) LOC count
- ▶ Socially rewarding turning pig ugliness into beauty

Distribution of developers and commits



Plenty of scope for shining in the project – with only 10+ commits.

Easy: fixing all SvStream operators ...

▼ So:

- ▼ `git clone git://anongit.freedesktop.org/libreoffice/core`

▼ Load up: `tools/inc/tools/stream.hxx`

- ▼ `SvStream& operator<<(sal_Int16 nInt16);`
- ▼ `SvStream& operator<<(sal_Int32 nInt32); ...`

▼ Remove / re-compile with:

- ▼ 'void WriteInt16(sal_Int16 nInt)' style methods

- ▼ Update all calling code & remove operator overload.

```
if( pPiece->IsUnicode() )
-     Strm() << c;
+     Strm().WriteInt32(c);
else
-     Strm() << (sal_uInt8)c;
+     Strm().WriteuInt8(c);
```

Significant: improves security & hackability

- ▶ Operator overloading / Magic type selection & inference
 - ▶ ripple/format problems from ~unrelated code change
- ▶ Nail security bugs from lame impl. of read operators:

```
SvStream& SvStream::operator>>(sal_uInt32& r)
{
    sal_uInt32 n = 0;
    READNUMBER_WITHOUT_SWAP(sal_uInt32, n)
    if (good())
    {
        if (bSwap)
            SwapULong(n);
        r = n;
    }
    return *this;
}
```

Ergo 'read' operator changes need some degree of auditing

Easy: using fsync where possible – a harder one

- ▼ The task is this:
 - ▼ `git clone git://anongit.freedesktop.org/libreoffice/core`
- ▼ Lots of detail in the hack:
 - ▼ https://bugs.freedesktop.org/show_bug.cgi?id=40607
- ▼ Checkout:
 - ▼ `sal/osl/unx/file.cxx (sal_oslSyncFile)`
 - ▼ Integrate a tweaked patch.
 - ▼ Call `fsync` only when it doesn't cripple the machine
- ▼ Find, and add `sal_oslSyncFile`
 - ▼ at last stage of document write.

Significant: why it's worth doing

- ▶ Only a few dozen lines of code produced
 - ▶ Linux only too of course.
- ▶ We have ~35 million Linux desktop users
- ▶ Somewhere, someone is about to swap their battery
 - ▶ **Knowing** they are safely on mains power ...
 - ▶ Only to discover that they are not as the yank it.
- ▶ Without this patch on ext4/btrfs ...
 - ▶ by-bye data ...
- ▶ With an 'obvious' 'fsync always' patch
 - ▶ Trash innumerable CDR's and judder audio etc.
- ▶ Help a friend in need ! ...

Easy: To create your own easy hacks ...

- ▼ Head to bugzilla:
 - ▼ https://bugs.freedesktop.org/enter_bug.cgi?product=LibreOffice
- ▼ Summary: “EasyHack:
- ▼ Component: “LibreOffice” or ...
- ▼ Description:
 - ▼ a **clear** set of code pointers
 - ▼ Enough detail to jump **immediately** into the code / task
 - ▼ ie. do the leg-work, a good EasyHack takes time to file.
- ▼ [Submit]
- ▼ **One more step:**
 - ▼ Add Whiteboard: see [list of names](#) or just use:
 - ▼ EasyHack,DifficultyBeginner,SkillCpp,TopicCleanup

Conclusions ...

Easy Hacks → Conclusions

- ▶ **Easy ! (or hard if you want)**
- ▶ **Significant**
 - ▶ they make a real difference
 - ▶ they strike a dis-proportionate blow for freedom
- ▶ **You can do them ...**
- ▶ Thanks for all of your help and support !
 - ▶ **we rely on you**, to prove the power of Freedom with us.
 - ▶ get people **excited** about Free Software they use ...
- ▶ **Please do get involved** and help us ...

Oh, that my words were recorded, that they were written on a scroll, that they were inscribed with an iron tool on lead, or engraved in rock for ever! I know that my Redeemer lives, and that in the end he will stand upon the earth. And though this body has been destroyed yet in my flesh I will see God, I myself will see him, with my own eyes - I and not another. How my heart yearns within me. - Job 19: 23-27